# CSE331: Microprocessor Interfacing and Embedded Systems

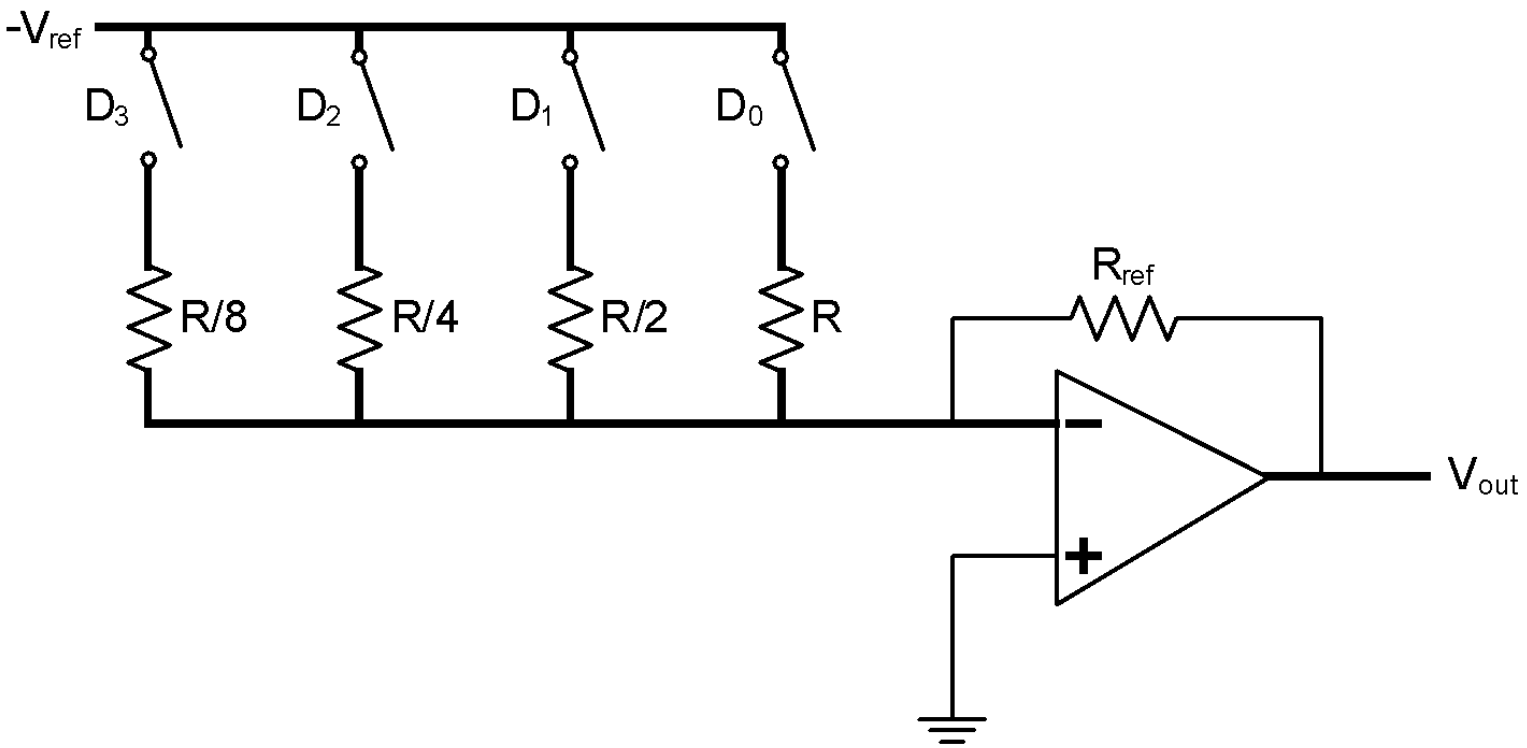
**Lecture#20: Digital-to-Analog Converter (DAC)**
**Analog-to-Digital Converter (ADC)**
**Chapter 20 and 21**

Courtesy: Mohammad A. Qayum, Ph.D.
ECE, NSU, Dhaka, Bangladesh

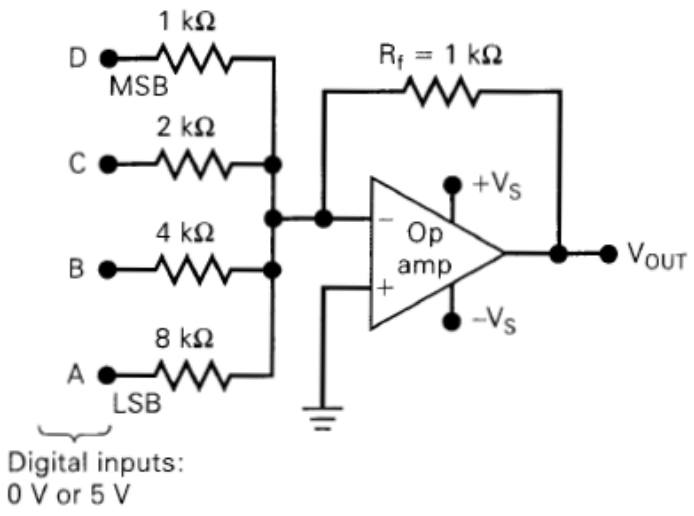
# DAC Implementations

- Pulse-width modulator (PWM)
- Binary-weighted resistor (We will use this one as an example)
- R-2R ladder (A special case of binary-weighted resistor)

# Binary-weighted Resistor DAC




$$V_{out} = V_{ref} \times \frac{R_{ref}}{R} \times (D_3 \times 2^3 + D_2 \times 2^2 + D_1 \times 2 + D_0)$$




(a)

Input code

| D | C | B | A | $V_{OUT}$ (volts) | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 1 | −0.625 | ← LSB |
| 0 | 0 | 1 | 0 | −1.250 | |
| 0 | 0 | 1 | 1 | −1.875 | |
| 0 | 1 | 0 | 0 | −2.500 | |
| 0 | 1 | 0 | 1 | −3.125 | |
| 0 | 1 | 1 | 0 | −3.750 | |
| 0 | 1 | 1 | 1 | −4.375 | |
| 1 | 0 | 0 | 0 | −5.000 | |
| 1 | 0 | 0 | 1 | −5.625 | |
| 1 | 0 | 1 | 0 | −6.250 | |
| 1 | 0 | 1 | 1 | −6.875 | |
| 1 | 1 | 0 | 0 | −7.500 | |
| 1 | 1 | 0 | 1 | −8.125 | |
| 1 | 1 | 1 | 0 | −8.750 | |
| 1 | 1 | 1 | 1 | −9.375 | ← Full-scale |

(b)

FIGURE 10-6 Simple DAC using an op-amp summing amplifier with binary-weighted resistors.

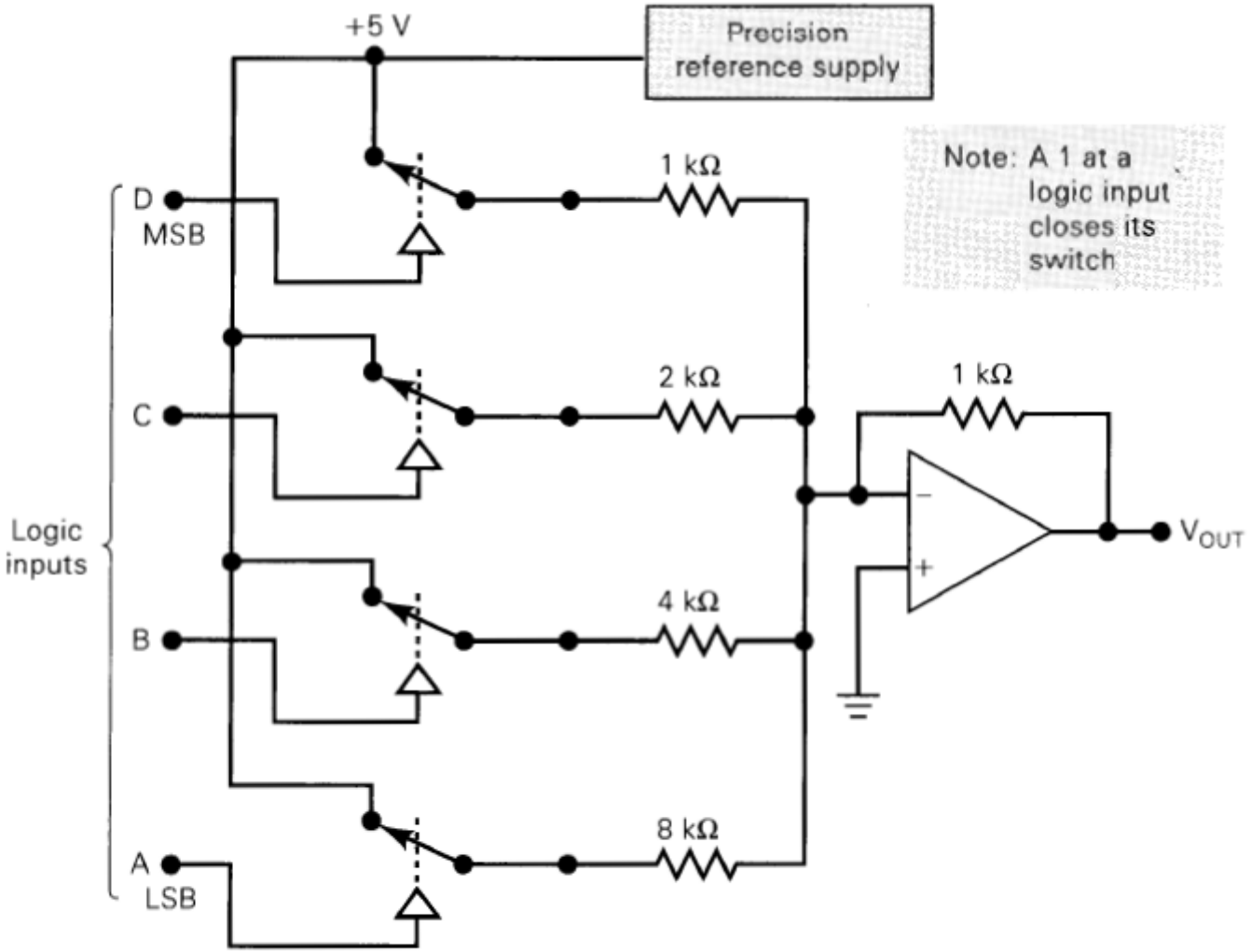



FIGURE 10-7 Complete four-bit DAC including a precision reference supply.

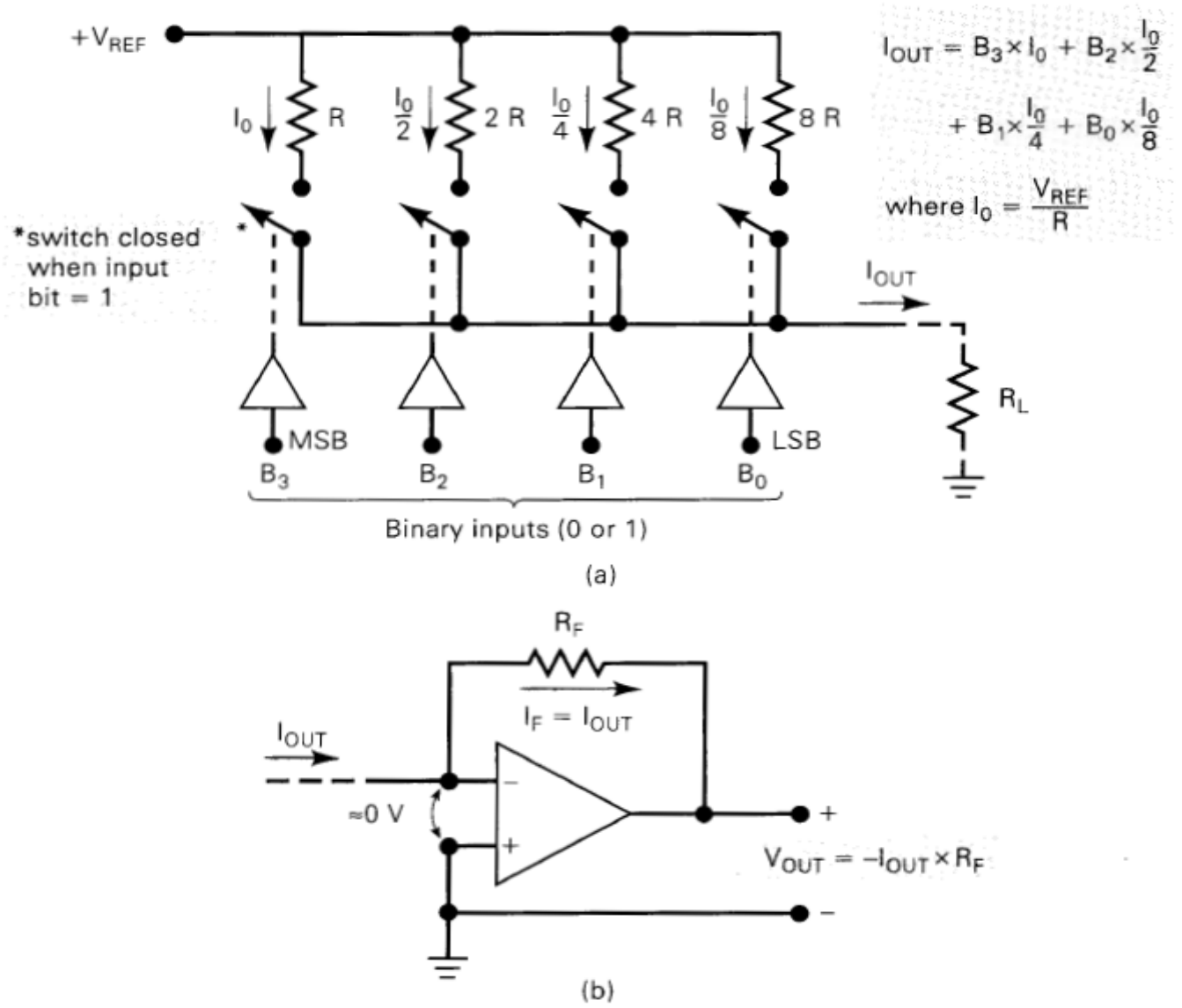



**FIGURE 10-8** (a) Basic current-output DAC; (b) connected to an op-amp current-to-voltage converter.

Assume that $V_{REF} = 10$ V and $R = 10$ kΩ. Determine the resolution and the full-scale output for this DAC. Assume that $R_L$ is much smaller than $R$.

**Solution**

$I_{OUT} = V_{REF}/R = 1$ mA. This is the weight of the MSB. The other three currents will be 0.5, 0.25, and 0.125 mA. The LSB is 0.125 mA, which is also the resolution.

The full-scale output will occur when the binary inputs are all HIGH so that each current switch is closed and

$$I_{OUT} = 1 + 0.5 + 0.25 + 0.125 = 1.875 \text{ mA}$$



Note that the output current is proportional to $V_{REF}$. If $V_{REF}$ is increased or decreased, the resolution and the full-scale output will change proportionally.

# Digital-to-analog converter (DAC)

- Converts digital data into a voltage signal by a N-bit DAC

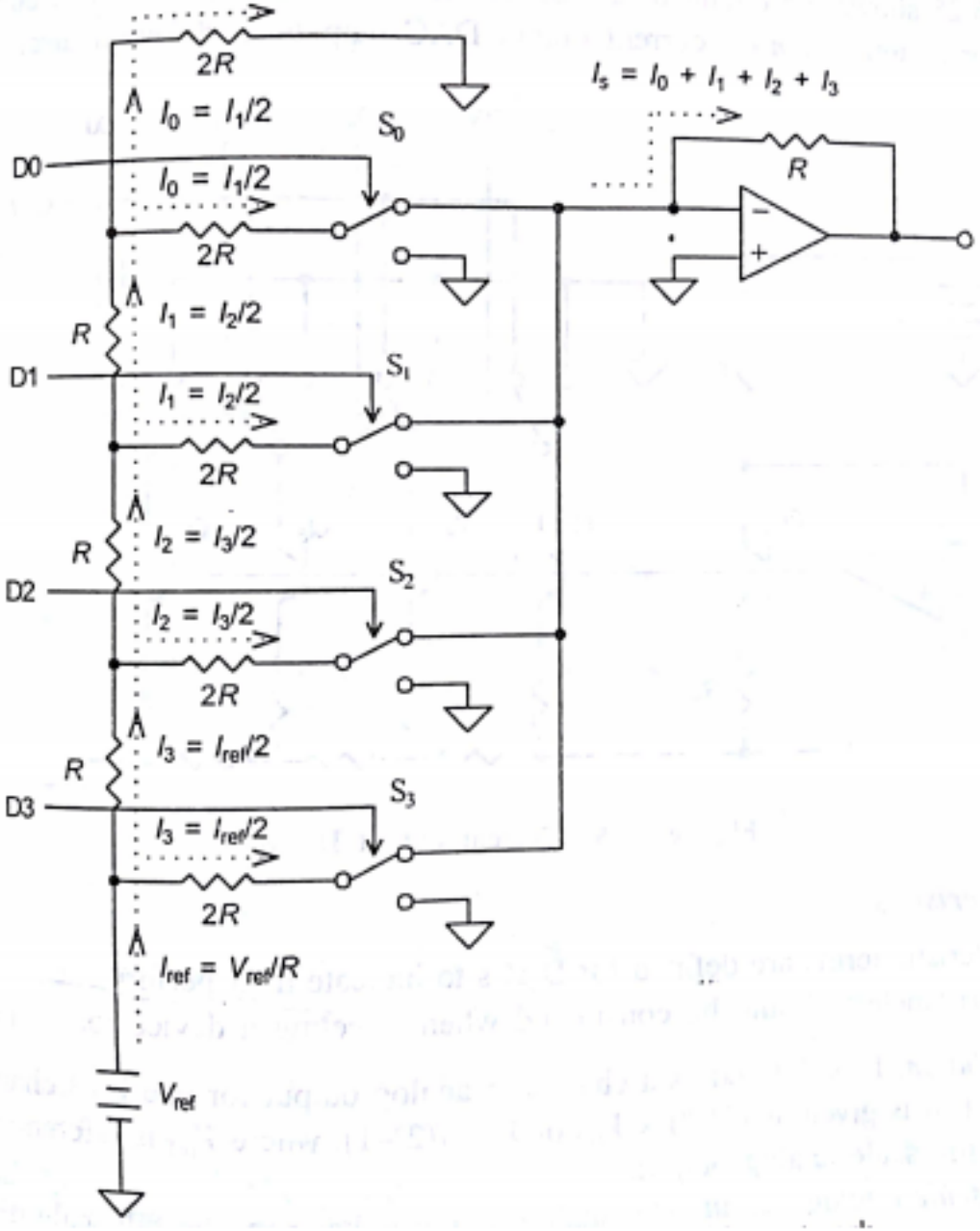
$$DAC_{output} = V_{ref} \times \frac{Digital\ Value}{2^N}$$

- For 12-bit DAC

$$DAC_{output} = V_{ref} \times \frac{Digital\ Value}{4096}$$

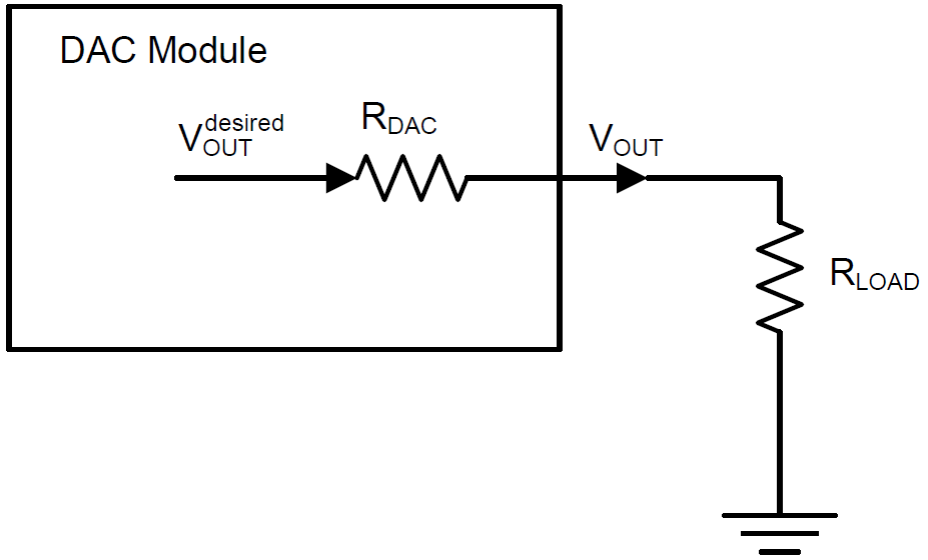
- Many applications:
  - digital audio
  - waveform generation
- Performance parameters
  - speed
  - resolution
  - power dissipation

R-2R ladder circuit, DAC



# Buffered Output

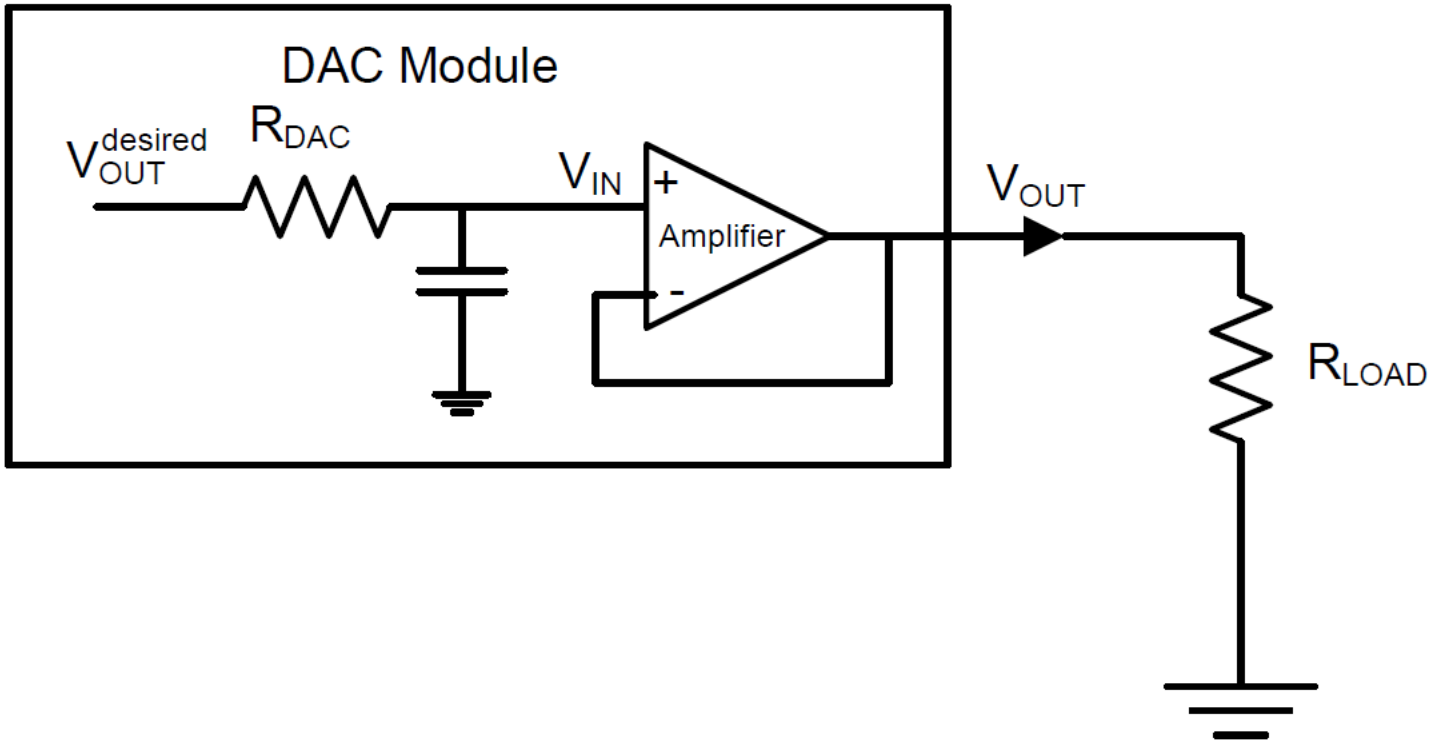
- Load Effect




$$V_{OUT} = \frac{R_{LOAD}}{R_{DAC} + R_{LOAD}} \times V_{OUT}^{desired}$$

# Buffered Output

- Use buffer to remove load effect




$$V_{OUT} \approx V_{OUT}^{desired}$$

We can use an internal output buffer to avoid the load impedance problem. As shown in Figure 21-6, the voltage output buffer is implemented by using an amplifier.

- The amplifier has a high input impedance (close to infinity) so that the impact of $R_{DAC}$ is diminished.
- The amplifier also has a low output impedance (close to zero) so that the effect of $R_{LOAD}$ is removed.

Thus, $V_{OUT}$ is kept close to the voltage desired.

$$V_{IN} = \frac{R_{IN,Amplifier}}{R_{DAC} + R_{IN,Amplifier}} \times V_{OUT}^{desired} \approx \frac{\infty}{R_{DAC} + \infty} \times V_{OUT}^{desired} = V_{OUT}^{desired}$$

$$V_{OUT} = \frac{R_{LOAD}}{R_{LOAD} + R_{OUT,Amplifier}} \times V_{IN} \approx \frac{R_{LOAD}}{R_{LOAD} + 0} \times V_{IN} = V_{IN} \approx V_{OUT}^{desired}$$

The output buffer can be enabled or disabled by the `DAC_CR_BOFF` bit in the control register `DAC_CR`.

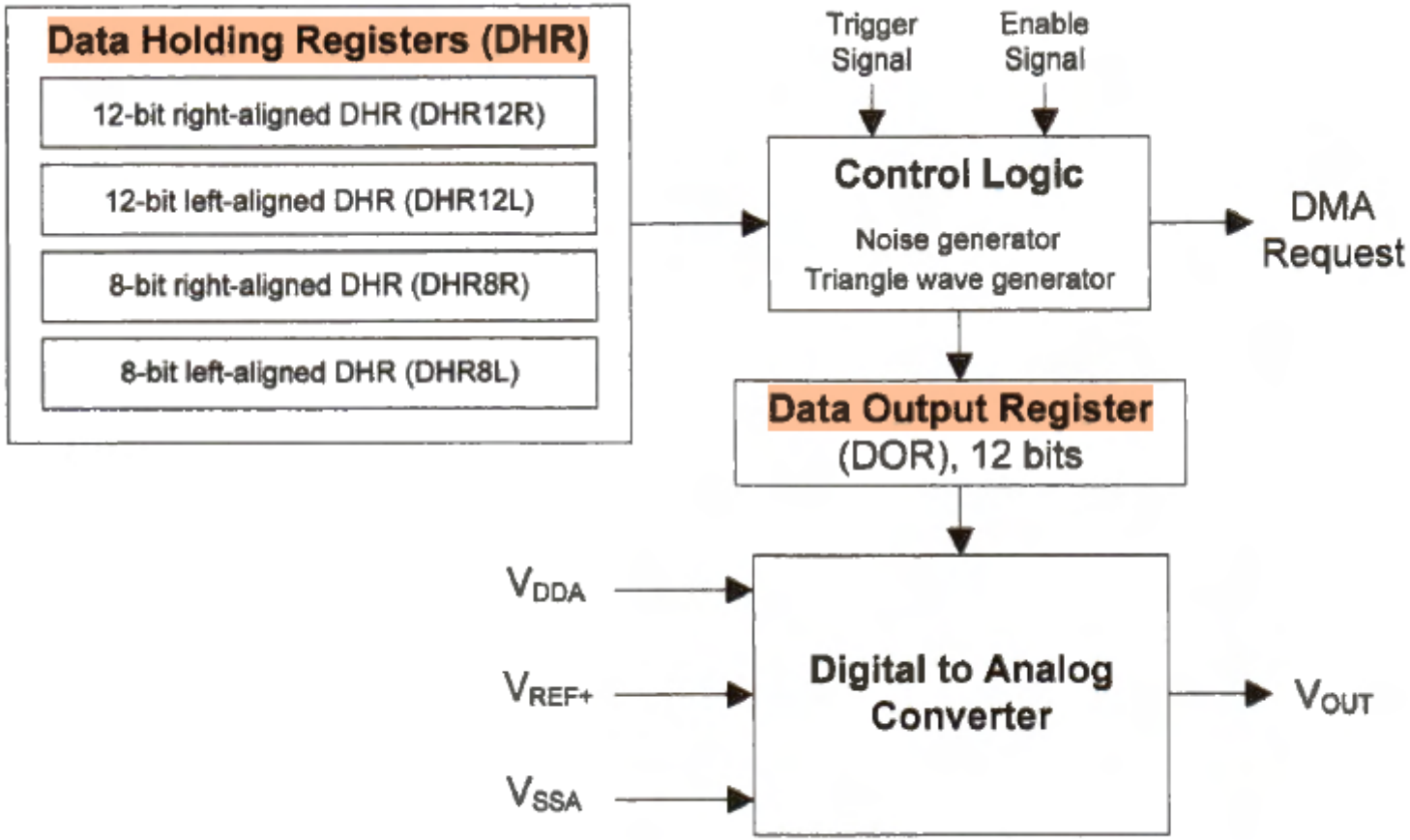
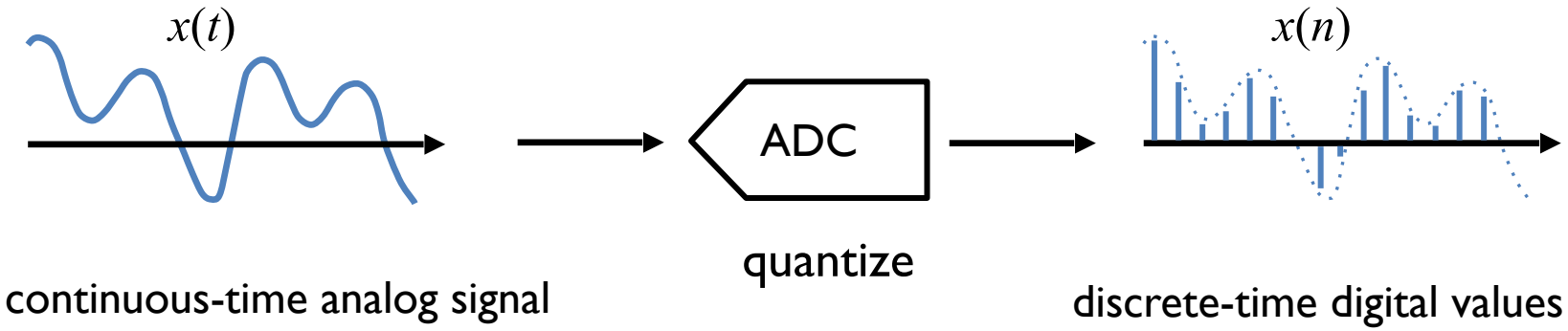



Figure 21-2. Digital to Analog Converter (DAC)

# Analog-to-Digital Converter (ADC)

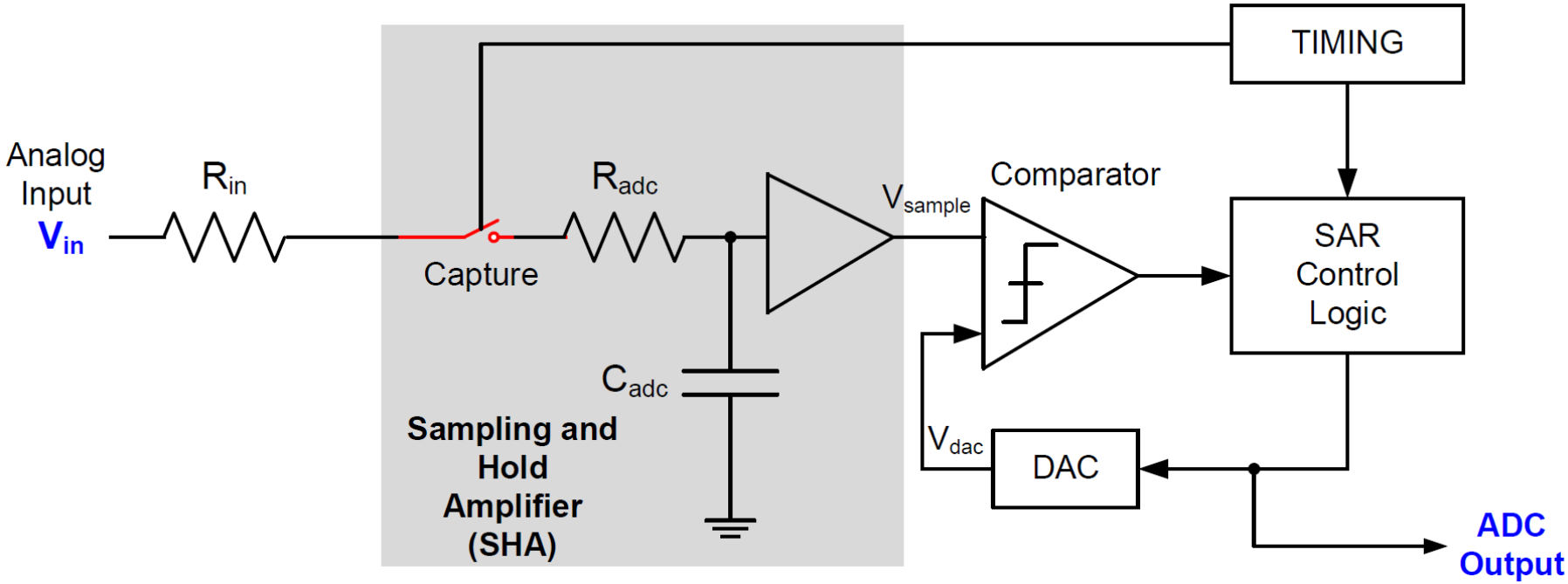
- ADC is important almost to all application fields
- Converts a continuous-time voltage signal within a given range to discrete-time digital values to quantify the voltage's amplitudes



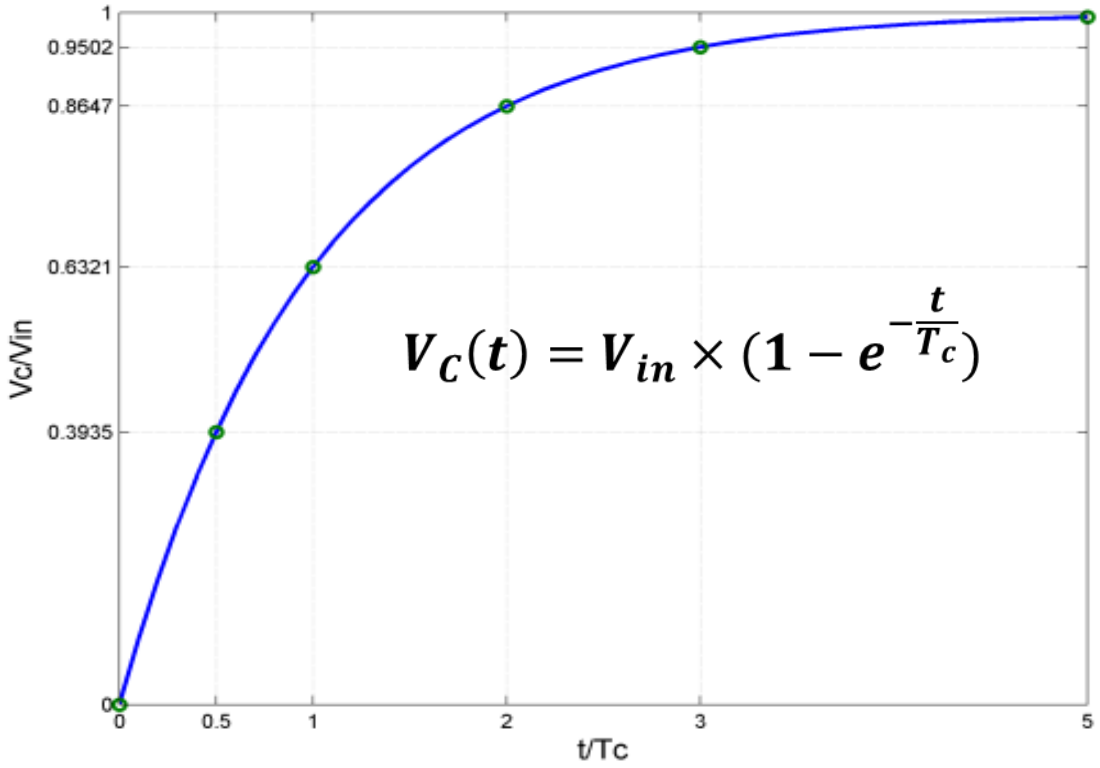
# Analog-to-Digital Converter (ADC)

- Three performance parameters:
  - sampling rate
  - resolution
  - power dissipation
- Many ADC implementations:
  - sigma-delta
  - successive-approximation
  - pipeline

# Successive-approximation (SAR) ADC



# Determining Minimum Sampling Time



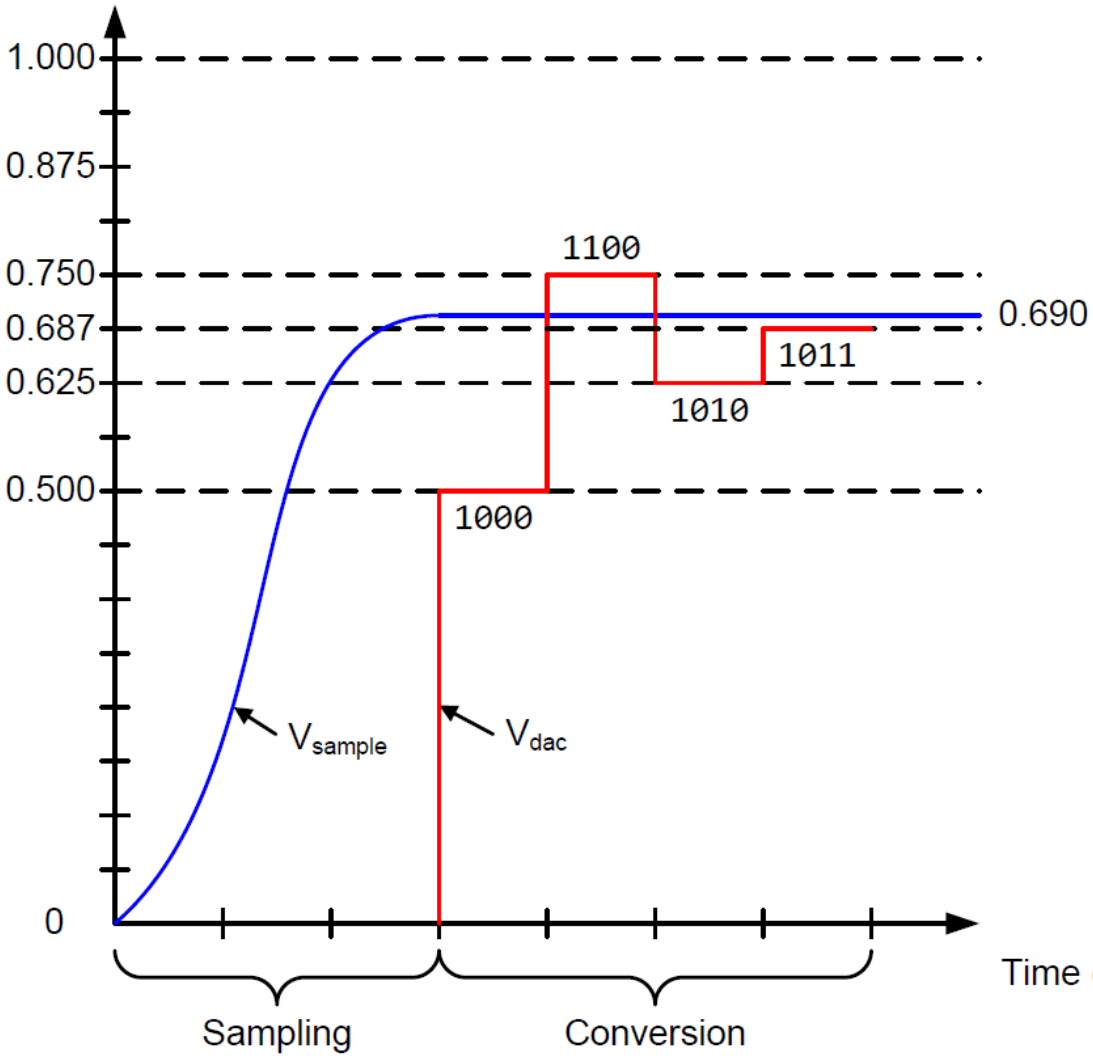

Sampling time is software programmable!

Larger sampling time → Smaller sampling error

Larger sampling time → Slower ADC speed

*Tradeoff*

# Successive-approximation (SAR) ADC



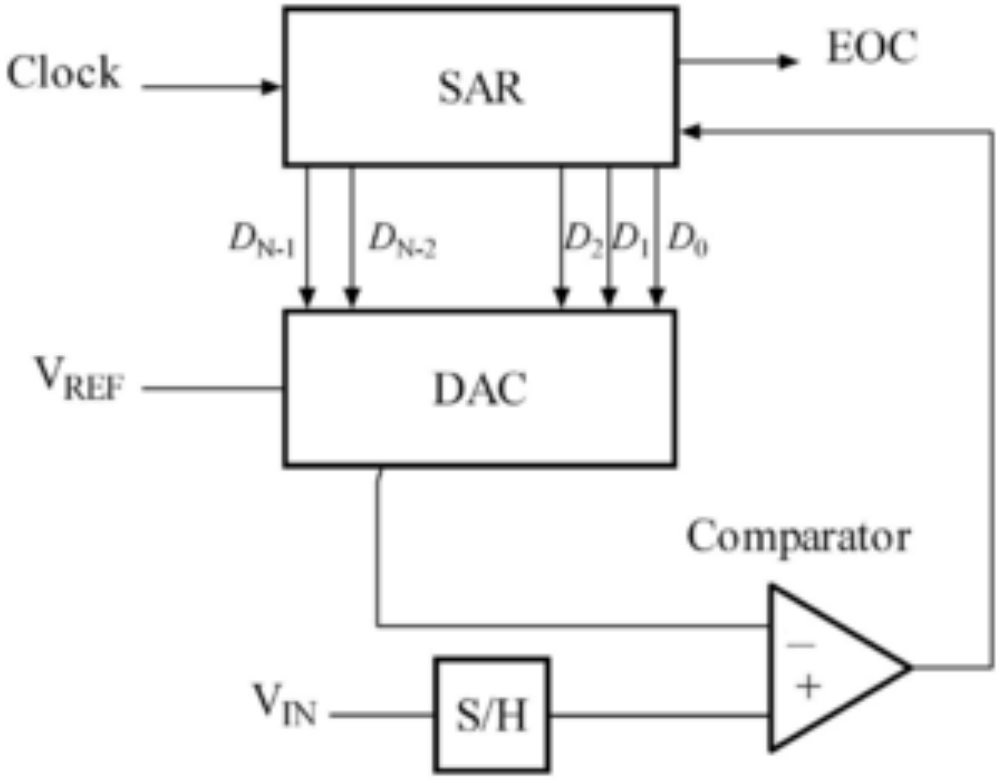

- Binary search algorithm to gradually approaches the input voltage
- Settle into ±½ LSB bound within the time allowed

$$T_{ADC} = T_{sampling} + T_{Conversion}$$

$$T_{Conversion} = \mathrm{N} \times T_{ADC\_Clock}$$

$T_{sampling}$ is software configurable

# SAR: Successive Approximation Register




The binary weights assigned to each bit, (Dx) starting with the MSB, are 2.5, 1.25, 0.625, 0.3125, 0.15625, 0.078125, 0.0390625, 0.01953125, 0.009765625, 0.0048828125. All of these add up to 4.9951171875, meaning binary 1111111111, or one LSB less than 5.

# ADC Conversion Time

$$T_{ADC} = T_{sampling} + T_{Conversion}$$

Suppose ADCCLK = 16 MHz and Sampling time = 4 cycles

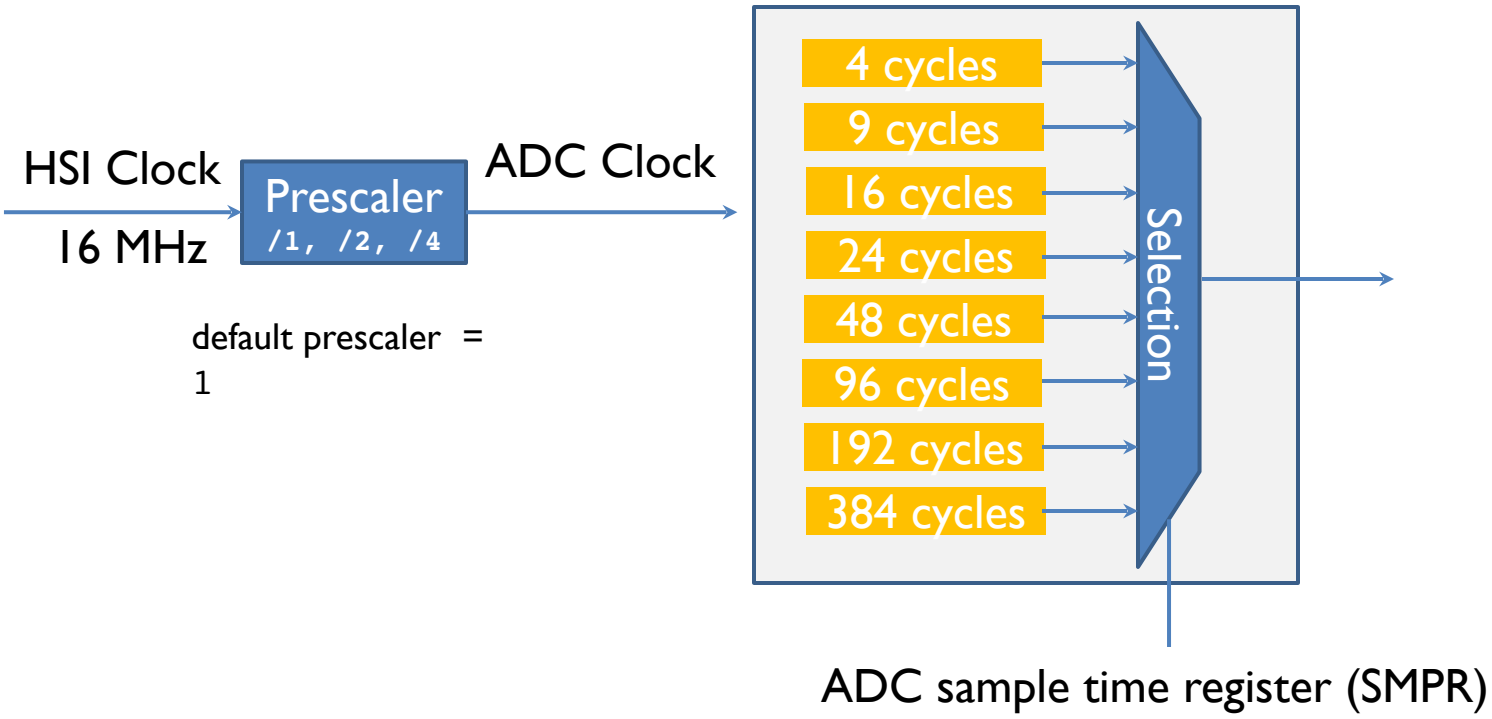
For 12-bit ADC

$$T_{ADC} = 4 + 12 = 16\ cycles = 1\mu s$$

For 6-bit ADC

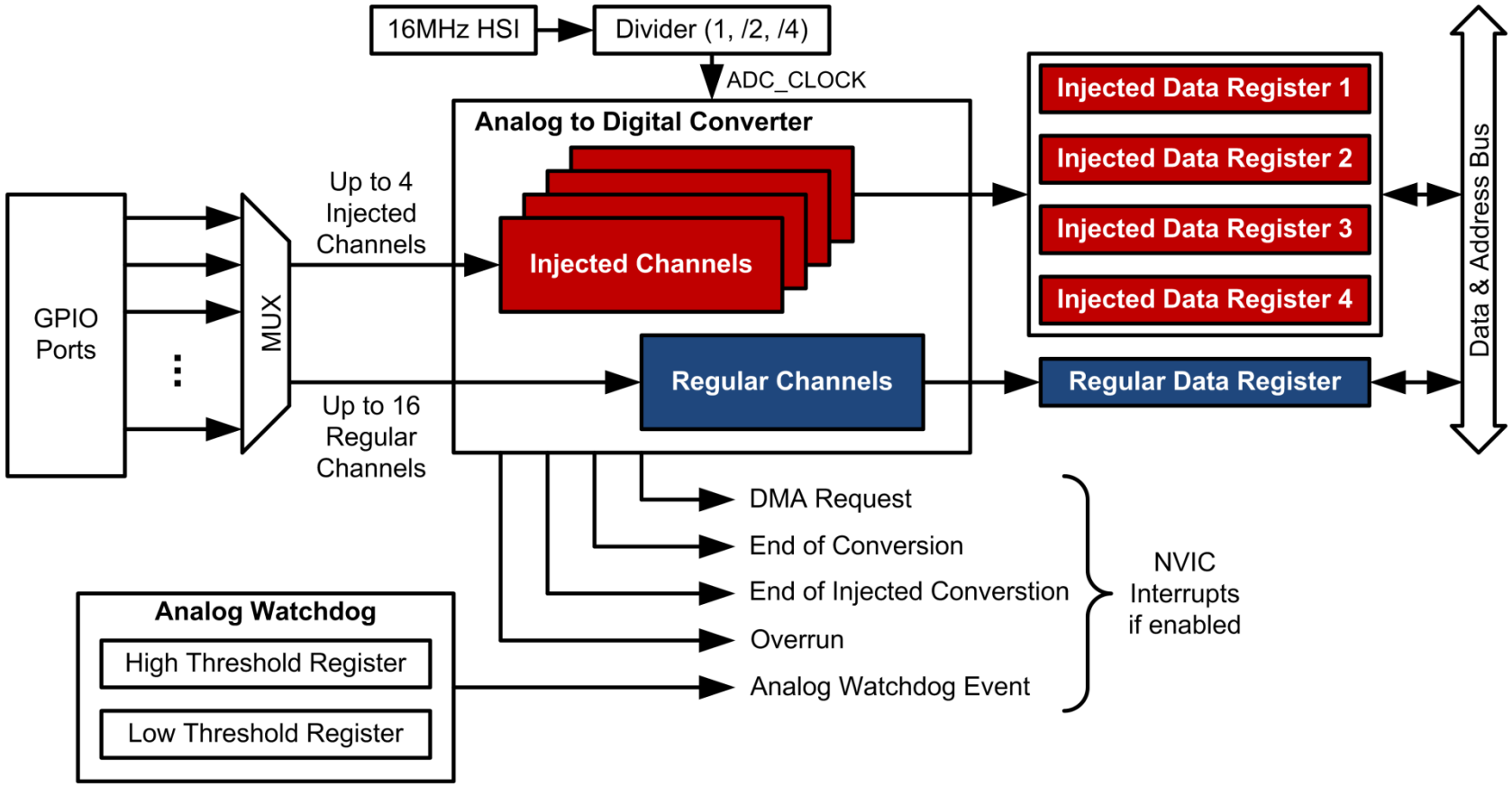
$$T_{ADC} = 4 + 6 = 10\ cycles = 625ns$$

# Programming ADC Sampling Time




default prescaler = 1

# ADC: Regular *vs* injected



Stepper motor

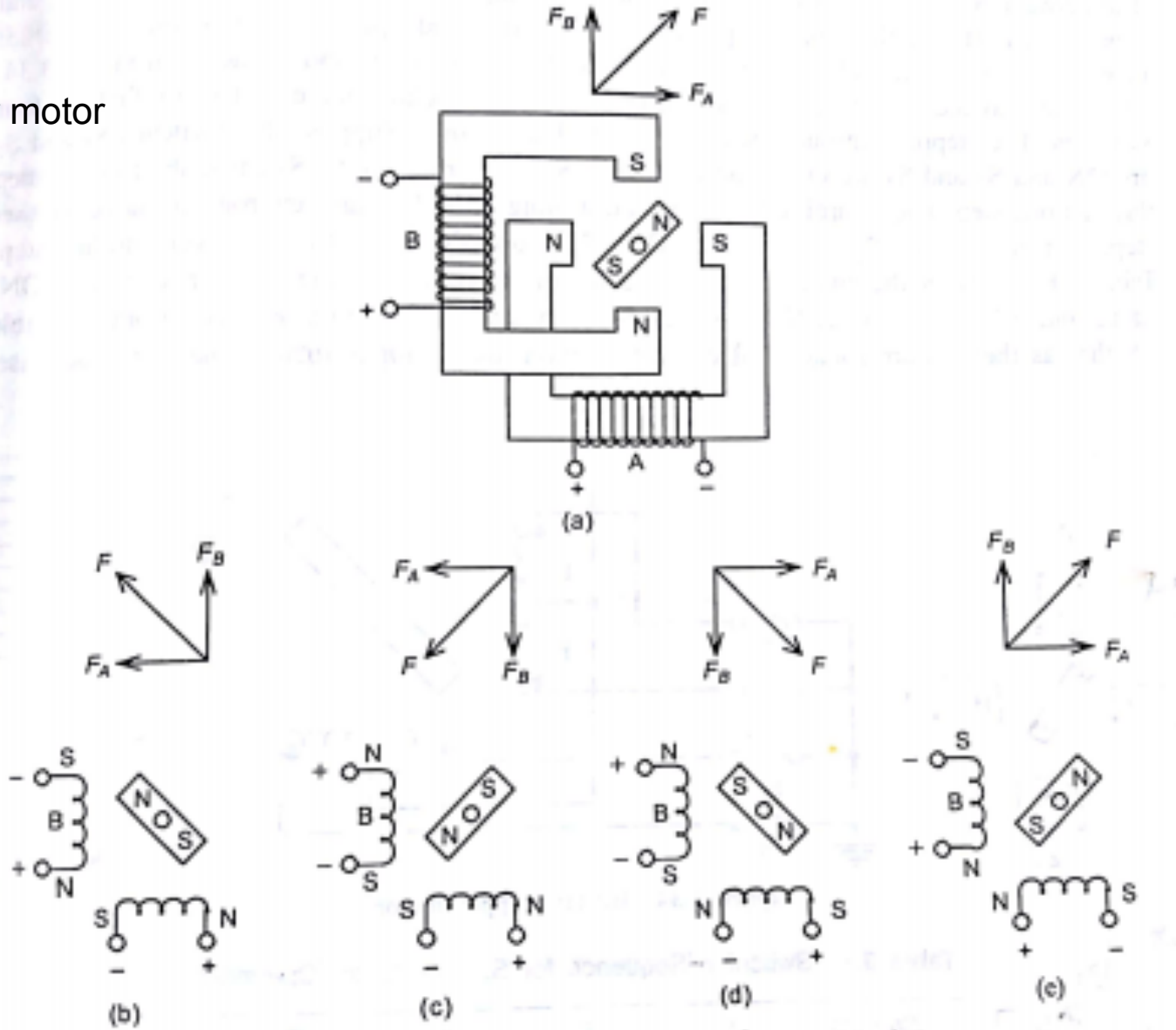



**Figure 3.33** Stepping through action of stepper motor.

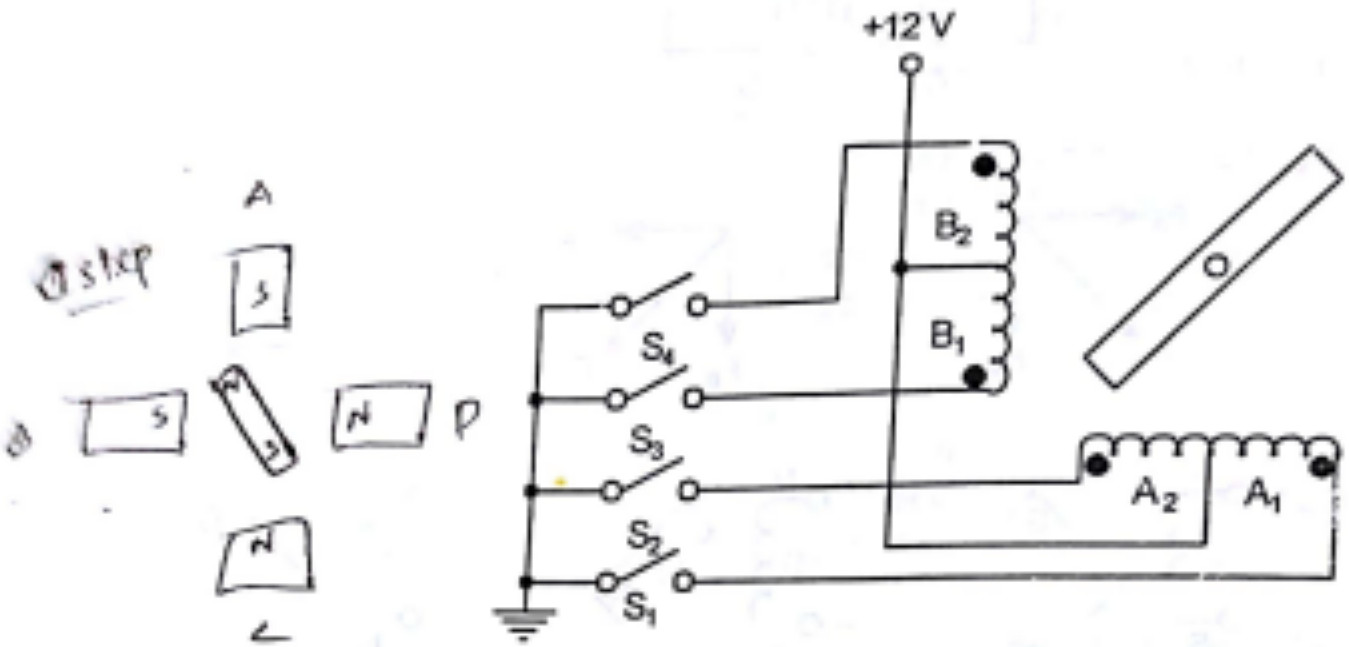



**Figure 3.34** Bifilar stepper motor.

**Table 3.8** Switching Sequence for Stepper Motor Operation

| Step No. | $S_4$ | $S_3$ | $S_2$ | $S_1$ |
|---|---|---|---|---|
| — | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 2 | 1 | 1 | 0 | 0 |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 0 | 0 | 1 | 1 |
| | A | B | C | D |

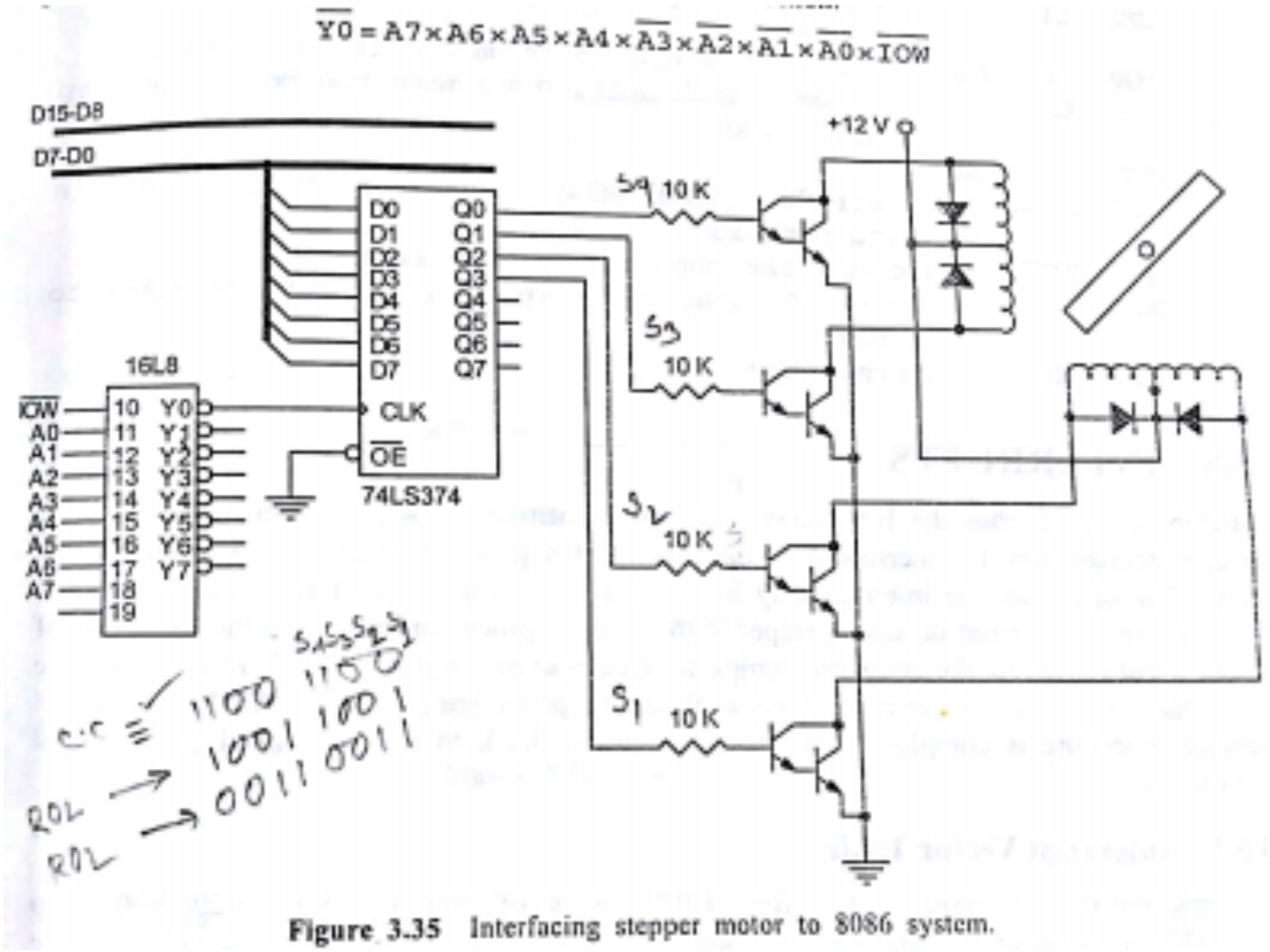



**Figure 3.35** Interfacing stepper motor to 8086 system.

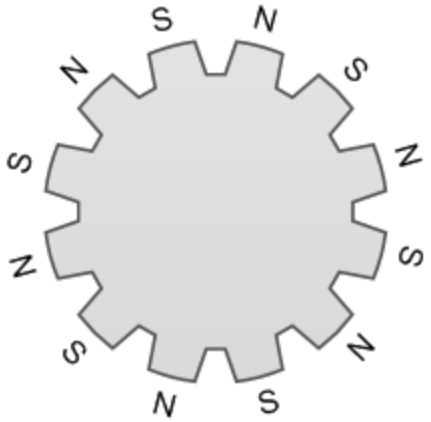
# Stepper Motor

$$Step\ Angle = \frac{360^{\circ}}{steps\ per\ revolution}$$

$$steps\ per\ revolution = P \times T$$

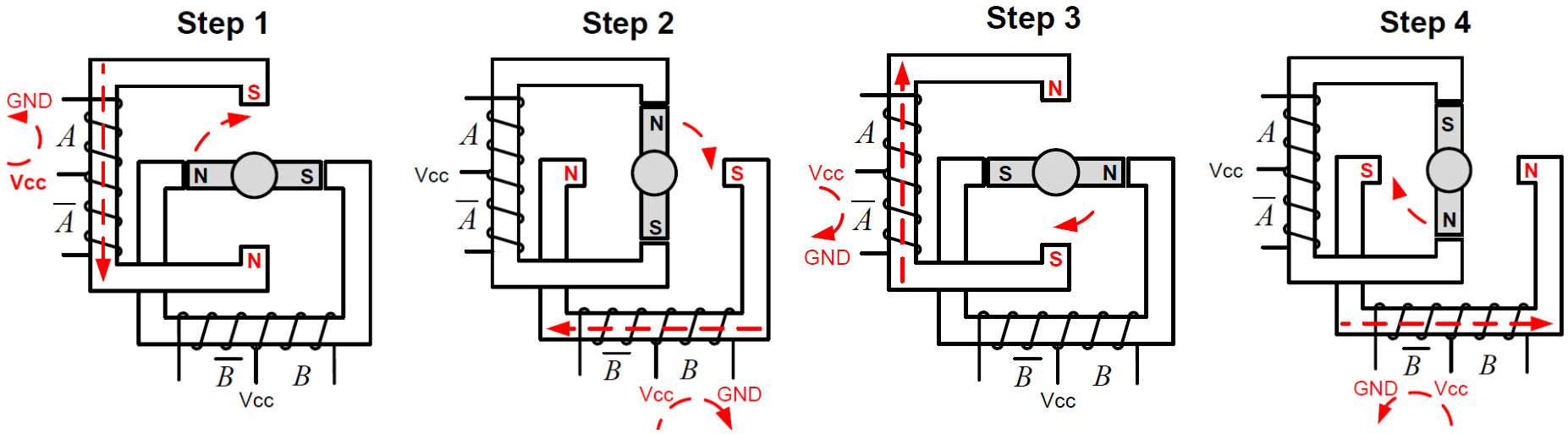
where $P$ is the total number of phases on the stator, and $T$ is the total number permanent-magnetic poles available on the rotor.

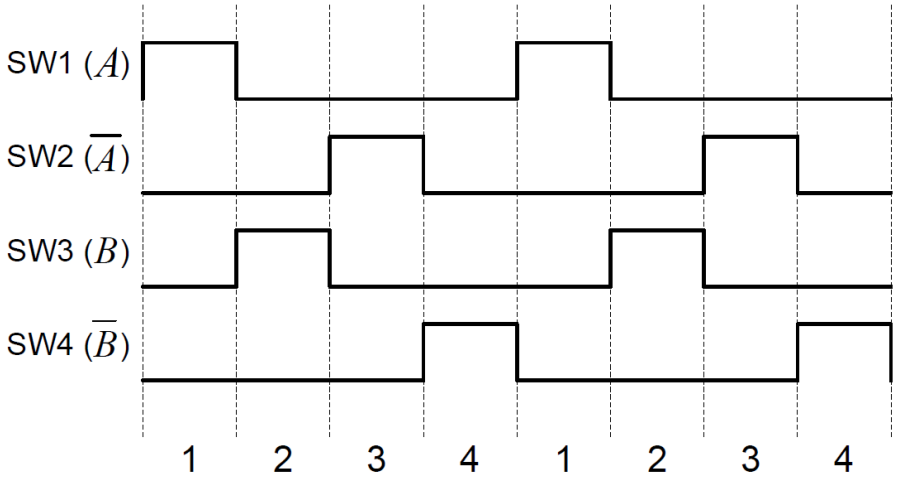

Rotor with only two poles




Rotor with 12 poles

# Wave Stepping



# 16.7 Driving Stepper Motor

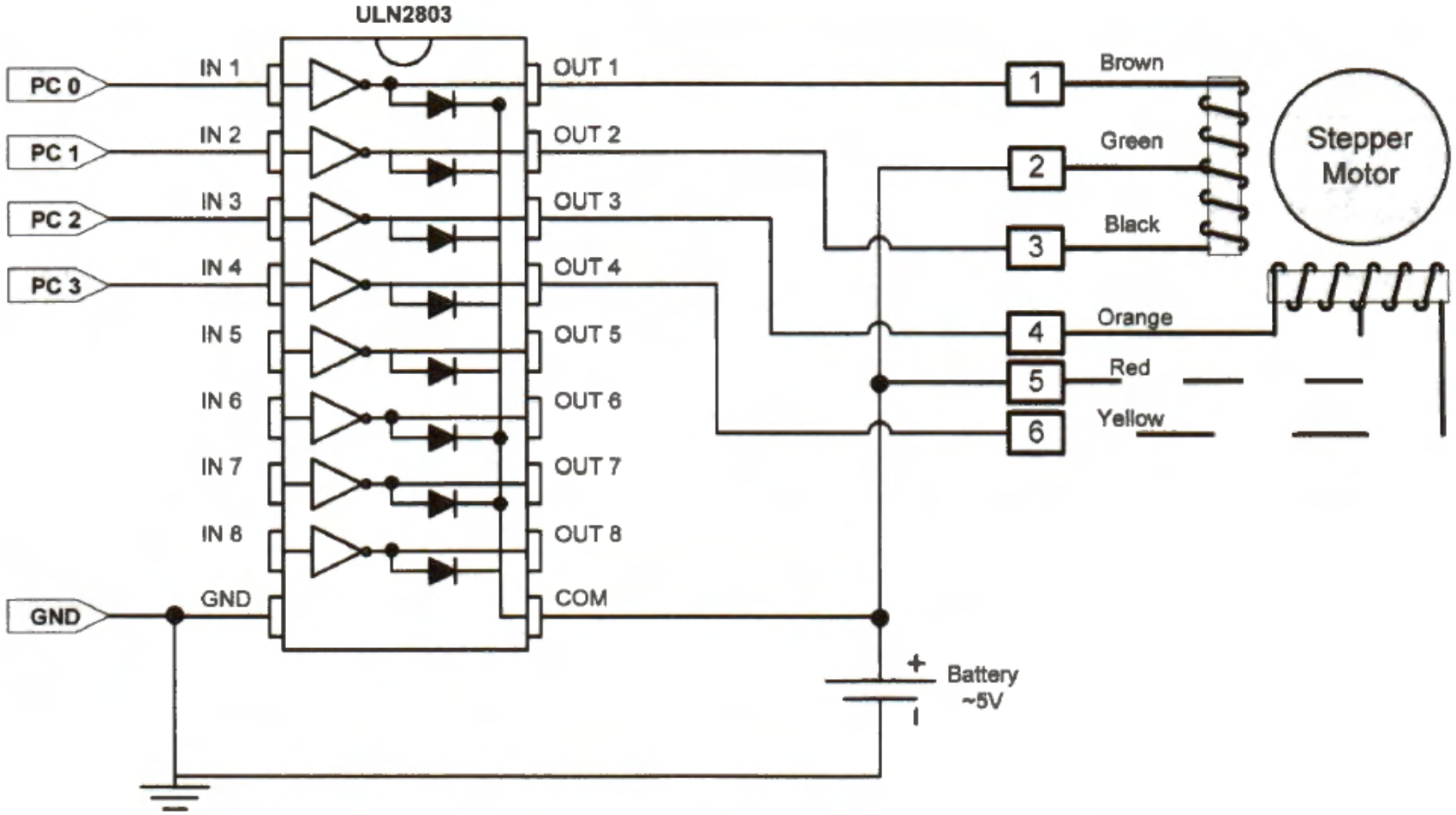



Figure 16-15. Connection diagram of driving a stepper motor